EXPLORING SELF-SUPERVISED REPRESENTATIONS FOR TEXT-DEPENDENT SPEAKER VERIFICATION

Anonymous submission to SLT2024 workshop

ABSTRACT

This paper presents our submission to the text-dependent speaker verification challenge 2024 (TDSV-2024). Deep embedding-based methods are widely used in text-independent speaker verification (TI-SV). However, they are less explored in text-dependent speaker verification (TD-SV) because labeled data for TD-SV is limited. Given the success of self-supervised models in downstream tasks with minimal labeled data, we investigate the use of self-supervised representations for deep embedding-based text-dependent speaker verification (TD-SV) systems. Additionally, we investigate the approach of combining separate TI-SV systems with password verification models, both using self-supervised representations, for the text-dependent speaker verification (TD-SV) task. First, the password verification model rejects trials with incorrect passwords. The remaining trials are then scored using the TI-SV system. Finally, our submission to the challenge achieved a minDCF of 0.029 in Track 1 (ranked 1^{st}) and 0.142 in Track 2 (ranked 3^{rd}).

Index Terms— Text-dependent speaker verification, self-supervised representations, password verification, WavLM.

1. INTRODUCTION

Speaker verification (SV) is the task of authenticating the claimed identity of a speaker from his/her voice characteristics. Based on the mode of verification, SV systems are divided into two types: 1) TI-SV and 2) TD-SV. The TI-SV system verifies only the speaker characteristics of the speaker, whereas the TD-SV system verifies both the speaker characteristics and the textual information. TI-SV systems are more flexible as they do not require the user to memorize the enrolled textual information, i.e., password. However, developing these systems is more challenging due to the phonetic mismatch between enrolled and test speech signals. As a result, TI-SV systems suffer from poor performance and are not recommended for deployment in high-security applications. The TD-SV system, which requires the user to say the specific password, is relatively more accurate than TI-SV systems. Most of the high-security applications prioritize TD-SV systems over TI-SV systems. Therefore, it is essential to investigate TD-SV under more realistic conditions by conducting challenges [1]. This paper reports our team submission to the TDSV-2024 challenge [2].

Initial approaches to SV relied on density estimation methods such as Gaussian Mixture Models (GMM-UBM) [3], I-vectors [4], etc. Later, with the advent of deep learning approaches, the development of SV systems has largely relied on deep speaker embedding extractors [5]. Deep speaker embedding extractors project the speech signal onto speaker discriminative space. Typically, these projections are obtained by training a speaker classification network using a huge number of background speakers' speech segments. Most recent research focused on developing more advanced speaker embedding extractors such as x-vector [6], SE-ResNet [7], ECAPA-TDNN [8], ERes2Net [9], etc. While this method significantly improved performance, training the network requires a huge amount of labeled speech data. Even then, these methods are investigated intensively since there are a lot of publicly available labeled data for TI-SV tasks [10–12].

On the other hand, TD-SV data are scarce since data with real passwords are not shared due to privacy reasons. As a result, there are only a few publicly available datasets [13, 14], and these databases are quite small, typically featuring around 100 speakers and ten different passwords. Consequently, the development of TD-SV systems has primarily relied on density estimation methods (GMM-HMM) [15, 16] and cascaded systems, where password verification and TI-SV are carried out sequentially. In addition, deep neural network-based TD-SV systems are also investigated in the literature. Still, most of them are customized for specific keywords [17–19], i.e.,"OK Google", "ni hao, mi ya", etc. Recent studies have also explored embedding-based approaches for developing more generic TD-SV systems. Most of these systems fine-tune TI-SV systems to suit TD-SV tasks [20, 21]. For example, authors in [20] investigated the phrase-dependent probabilistic linear discriminant analysis models to adapt TI-SV to TD-SV. The work in [21] considers classifying speaker and passwords in a multitasking manner or considers classes as speaker-sentence pairs [21], i.e., each password spoken by a speaker is considered a separate class. Since the training data is less, the performance of these systems is poor. Given the success of self-supervised representations, especially for low-scale training data tasks, this work investigates the usage of self-supervised representations for TD-SV tasks.

The remainder of the paper is structured as follows: Section 2 introduces the embedding extractors used in this study.

Section 3 covers the datasets and experimental details. Sections 4 and 5 analyze the results and conclude the paper.

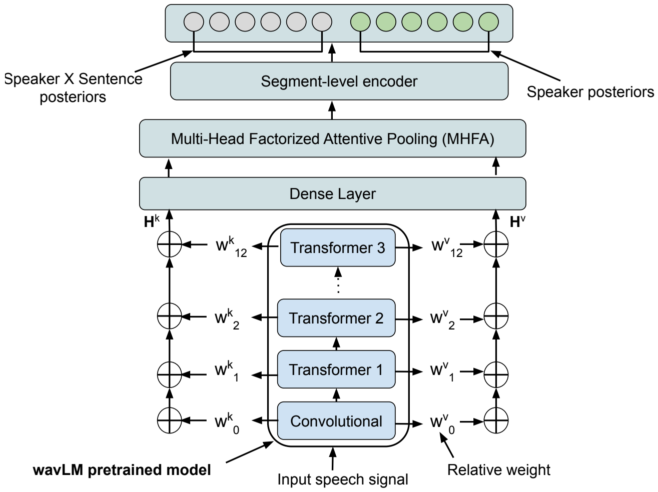

Fig. 1: Text-dependent speaker classifier. For Track 1, both speaker and speaker-sentence labels are included, and the system is called TD-SV-FreeText. In contrast, Track 2 uses only speaker labels, and the system is named TD-SV.

2. METHOD

2.1. Pretrained self-supervised model: WavLM

Over recent years, self-supervised models have emerged as a potential standard for representation extractors. Self-supervised pretraining aims at learning slowly varying features through predictive coding or reconstruction loss that does not require labeled data [22–25]. Hence, this pretraining can be performed using a large amount of unlabeled speech data. Once the pretraining is completed, slowly varying features can be extracted from the pretrained model and utilized for speech-processing tasks. Recent works have shown that self-supervised representations are useful for the majority of downstream tasks [26], i.e., speech recognition, speaker recognition, language identification, etc. One of the main advantages of self-supervised pretraining is that the extracted representations can be used to develop the downstream task even with a very small amount of labeled data. Among the existing self-supervised models, the WavLM model [25], which optimizes the masked prediction loss along with the denoising loss, is shown to be relatively superior for downstream tasks. In this work, we explored the usage of the pretrained WavLM model in the context of the TD-SV task. The pretrained WavLM model considered in this work is available publicly at `https://github.com/microsoft/unilm/tree/master/wavlm`.

2.2. TI speaker embedding extractor

The SV system in this work projects the speech signal onto a speaker discriminative space and compares the resultant projections, known as speaker embeddings, using the back-end scoring module. In this work, speaker embeddings are obtained from the penultimate layer activation of a speaker classification network trained on a large set of background speakers. A typical speaker classification network comprises three modules: 1) Frame-level encoder, 2) Pooling layer, and 3) Segment-level encoder. The frame-level encoder takes a variable-length speech signal as input and extracts high-level representations from it. The pooling layer obtains fixed dimensional embedding from the variable length frame-level representations. This fixed dimensional embedding is mapped to speaker posteriors using a couple of dense layers, typically referred to as segment-level encoders. We consider ERes2Net [9] as the baseline architecture. This baseline system uses Res2Net layers [27] to build up the frame-level encoder. This work replaces this frame-level encoder with the WavLM model pretrained in a self-supervised manner. The architecture of the modified speaker classification network is shown in Figure 1. The frame-level feature extractor is replaced with the pre-trained WavLM model. This model comprises a convolutional module followed by 12 transformer encoder layers. Following the literature [28], a weighted combination of layer representations is considered as frame-level features.

$$\mathbf{H}^v = w_0^v\mathbf{H}_0 + w_1^v\mathbf{H}_1 + ... + w_{12}^v\mathbf{H}_{12} \tag{1}$$

Where $\{w^v\}_0^{12}$ are the relative weights assigned to the corresponding layer representations $\{\mathbf{H}\}_0^{12}$. $\{\mathbf{H}\}_0$ is the output of the convolutional module, and $\mathbf{H}_1^{12}$ are the outputs of the 12 transformer encoder layers in the WavLM model. $\mathbf{H}^v$ are the frame-level representations obtained from the WavLM model. These frame-level representations are temporally aggregated to obtain a fixed dimensional embedding. One of the naive approaches is to take the mean and standard deviation across time. However, this naive approach assumes all the frame-level representations are equally important for speaker discrimination. This standard pooling layer is replaced with a self-attentive pooling layer that gives relative importance to different frame-level representations. While there are different varieties of self-attention layers [29, 30], multi-head factorized attentive pooling (MHFA) [25] is shown to work well for text-independent speaker verification systems. Authors in [25] claim that MHFA clusters frame-level representations into acoustic units discovered by the transformer model. The frame-level representations are pooled within each cluster to get cluster-specific fixed dimensional embeddings. Cluster-specific fixed dimensional embeddings are concatenated to get overall fixed dimensional embedding. Cluster assignments can be obtained by processing frame-level representations through a Dense layer followed by softmax activation function along the temporal dimension. However, they assume that frame-level representations need to capture speaker information and may not be well suited for getting cluster assignments. Hence, a separate set of weights is assigned to layer representations to get the Key matrix. This Key matrix is used to obtain the cluster assignments as below.

$$\mathbf{H}^k = w_0^k\mathbf{H}_0 + w_1^k\mathbf{H}_1 + ... + w_{12}^k\mathbf{H}_{12} \tag{2}$$

$$\mathbf{A} = softmax(\mathbf{H}^k\mathbf{Q}) \tag{3}$$

$$\mathbf{s}_z = \sum_{t=1}^{T}\mathbf{A}[z,t]\mathbf{H}^v[t] \tag{4}$$

$$\mathbf{s} = Concat(\mathbf{s}_1, \mathbf{s}_2, ..., \mathbf{s}_Z) \tag{5}$$

$\mathbf{H}^k$ is the Key matrix, $\mathbf{A} \in \mathcal{R}^{Z\times T}$ gives the cluster assignments with Z and T being the number of clusters and frames, respectively. $\mathbf{Q}$ is the weight matrix in the Dense layer. $\mathbf{s}_z$ is z^{th} cluster specific fixed dimensional embedding. Overall fixed dimensional embedding **s** is obtained by concatenating cluster-specific fixed dimensional embeddings. Authors in [25] claimed that the clusters discovered by $\mathbf{A}$ correlate with phonetic units. The system compares frame-level representations under the same phonetic unit and provides improvement to the extracted speaker embedding. However, they do not provide sufficient evidence for it. This work investigates MHFA in the context of text-dependent speaker verification and aims to understand the clusters.

Finally, the network weights are optimized to minimize the classification loss. Once the network is trained, the activation of the penultimate layer is used to obtain the speaker embedding. This work calls this embedding TI speaker embeddings as it aims to capture only the speaker information and does not highlight password information. TI speaker embeddings are extracted for registered and test voices and compared using a cosine similarity function.

2.3. TD speaker embedding extractor

This work modifies the labels in the classification network to highlight the password information in the extracted speaker embedding. Specifically, it considers every speaker's password as a separate class. For instance, if there are ten passwords and 1620 speakers in the training data, the text-dependent speaker classifier considers 16200 classes, whereas the text-independent speaker classifier considers only 1620 classes. The architecture of the text-dependent speaker classifier is shown in Figure 1. As observed from the figure, the architecture is the same as a text-independent classifier except for the number of classes. This classification network is trained with labeled text-dependent data. After training the network, the output from the segment-level encoder is extracted to obtain the speaker embedding. This speaker embedding is referred to as TD speaker embedding as it aims to capture both speaker and password information.

2.4. Cascaded system approach for TD-SV

The objective of TD-SV is to verify both password and speaker information simultaneously. While the text-dependent speaker embedding extractor method works well to verify password and speaker information simultaneously, we observe that the system is more biased toward speaker verification. Hence, we aim to investigate whether adding a password verification module before either a text-dependent or text-independent speaker embedding extractor can help reduce password verification error. The cascaded system approach first verifies the password information. The trials with different passwords are rejected before evaluating the text-dependent or text-independent speaker embedding extractor method. While this method is more intuitive for the combination of password verification and the TI-SV system, we observe that it also improves the performance of the text-dependent speaker embedding extractor method.

3. DATASETS & EXPERIMENTAL DETAILS

The performance of the proposed TD-SV system is evaluated using the TDSV 2024 challenge [2]. Data is taken from the DeepMine speech processing database [31, 32]. This challenge has two tracks. The first track focuses on the case where the users are restricted to use only the passwords in the training data. The second track relaxes this assumption and allows the users to use their own passwords irrespective of the training data. *Track 1:* This track training data consists of multiple repetitions of 10 passwords, 5 English and 5 Persian, spoken by 1620 speakers. Along with the 10 password data, free text spoken by the speaker is also provided to enhance the speaker modeling. The same ten passwords are used to evaluate the performance of the system. Three repetitions of each password are provided to register the voice, and the goal is to compare this registered voice with the single test speech signal. Since the system is text-dependent, it should verify both speaker and password information. *Track 2:* Since this track focuses on evaluating the ability of the TD-SV system to use unseen passwords, ten passwords are divided into two sets. One set having six passwords is used as training data. Another set with the remaining four passwords is used as the evaluation data. Training data also includes free text spoken by the speaker. During the evaluation, three repetitions of the password and free text spoken by each speaker were provided to register the voice. This registered voice is compared with the test speech signal to authenticate the speaker and password. Development data is provided for both tracks to identify the best suitable hyper-parameters empirically. We are not supposed to use development data for training the networks.

3.1. Experimental details

This work examines two advanced text-independent speaker embedding extractors. The ERes2Net architecture is acknowledged as a top system for text-independent speaker verification (TI-SV) using traditional acoustic features. However, the most prominent TI-SV system is the WavLM model combined with a backend classifier. The ERes2Net architecture and its training methodology follow the approach described in [9], while the WavLM-based TI-SV system follows the approach outlined in [25]. WavLM-based TI-SV system contains a pretrained WavLM-Base+ module combined with the backend classifier. This WavLM-Base+ module comprises a convolutional module and 12 transformer encoder layers, as shown in Figure 1. 94k hours of unlabeled speech data are used to pre-train this module. It is available for download at `https://github.com/microsoft/unilm/tree/master/wavlm`. During the training of both TI and TD speaker embedding extractors with the WavLM encoder, the parameters of the WavLM encoder are also fine-tuned to reduce the classification loss. Further details on TD-SV training and evaluation are available in our open-source implementation at `https://github.com/SpeechPublications/tdsv2024`.

3.1.1. Implementation details of password verification

The password classification network uses self-supervised representations from the pre-trained WavLM model as input. It then classifies passwords using a simple backend module that includes a self-attentive pooling layer, a segment-level encoder, and a logistic regression layer. After training, password embeddings are extracted from the segment-level encoder's outputs. We can compare these embeddings using cosine similarity scores and reject trials with scores below a certain threshold. This threshold can be determined empirically through the evaluation of development data.

3.1.2. TI-SV system

Table 1: EER of TI-SV systems evaluated on VoxCeleb1 test data.

TI-SV system	EER (%)
ERes2Net	0.97
WavLM-MHFA	0.96

ERes2Net and WavLM are trained using the VoxCeleb2 development dataset, which comprises approximately 2000 hours of speech from 5994 speakers [11]. The weights of this network are optimized to minimize the classification loss, additive angular margin softmax loss [33]. Performance of these TI-SV systems evaluated on VoxCeleb1 test data [10] is shown in Table 1. We can see that both the systems give almost equal performance, 0.97 % and 0.96 % EER, as the training dataset is large.. These systems are further fine-tuned using the in-domain challenge data, and their performance is reported in Table 2 under the heading "TI-SV system." Following the challenge evaluation plan [2], distinct TI-SV systems are developed for Track 1 and 2 using their respective in-domain training data.

3.1.3. TD-SV system

Following the challenge's rules, text-dependent speaker embedding extractors are trained independently for Tracks 1 and 2. Since training data comprises ten passwords and free text spoken by a speaker, we aim to study whether using free text in the training text-dependent speaker embedding improves performance. As shown in Figure 1, we have considered both speaker labels and speaker-password labels while training the text-dependent speaker classification. All the utterances of free text are assigned their speaker labels. Utterances of passwords spoken by a speaker are assigned with their speaker-password label. The TD-SV system, which was developed with the inclusion of free-text speaker labels, is called TD-SV-FreeText. We empirically observed that TD-SV-FreeText performs relatively better than TD-SV for Track 1, while TD-SV-FreeText performs worse for Track 2. Therefore, we used TD-SV-FreeText for Track 1 and the TD-SV system for Track 2, as reported in Table 2.

3.1.4. Password classifier

Track 1 of the challenge focuses on seen passwords in the evaluation, while Track 2 focuses on unseen passwords. The training data for Track 1 includes all ten passwords, whereas the training data for Track 2 includes only six passwords. Therefore, two password classification networks are trained using the 10-password Track 1 and 6-password Track 2 training data. WavLM self-supervised representations are used as inputs to the password classifier.

4. RESULTS & ANALYSIS

Table 2 presents the TD-SV performance on both Track 1 and Track 2 evaluation data. Two performance metrics, equal error rate (EER) and minimum detection cost function (minDCF), are provided. Table 2 presents the overall text-dependent performance and also provides performance details for different test conditions. For example, when evaluating the overall text-dependent performance, the trials encompass all possible combinations, such as Target-correct vs. Imposter-correct (TC-vs-IC) and Target-correct vs. Target-wrong (TC-vs-TW). In this context, TC-vs-IC refers to trials with different speakers but the same password, while TC-vs-TW refers to trials with the same speaker but different passwords. Good performance on TC-vs-TW indicates that the system effectively discriminates between passwords, while good performance on TC-vs-IC indicates that the system effectively discriminates between speakers.

Table 2: The table reports the EER / minDCF of systems evaluated on tdsv2024 challenge evaluation data

System No	Model	Track 1 Evaluation			Track 2 Evaluation		
	-	Overall	TC-vs-TW	TC-vs-IC	Overall	TC-vs-TW	TC-vs-IC
TI-SV system							
1	ERes2Net	15.7 / 0.880	36.3 / 0.884	2.53 / 0.100	17.0 / 1.000	49.0 / 1.000	1.77 / 0.797
2	WavLM	14.7 / 0.084	34.3 / 0.974	1.59 / 0.051	17.45 / 1.000	51.31 / 1.000	0.83 / 0.032
Track 1: TD-SV-FreeText system, Track 2: TD-SV system							
3	ERes2Net	2.97 / 0.113	4.39 / 0.154	2.20 / 0.083	5.67 / 0.265	8.77 / 0.378	4.48 / 0.206
4	WavLM	1.58 / 0.044	2.07 / 0.054	1.18 / 0.034	3.22 / 0.161	6.25 / 0.263	1.56 / 0.056
Password rejection & TI-SV system							
5	ERes2Net	2.35 / 0.093	0.37 / 0.006	2.57 / 0.101	4.21 / 0.375	12.75 / 0.999	3.48 / 0.099
6	WavLM	1.54 / 0.048	0.29 / 0.005	1.64 / 0.051	3.43 / 0.346	12.95 / 1.000	2.88 / 0.055
Password rejection & TD-SV system							
7	ERes2Net	2.02 / 0.077	0.44 / 0.011	2.21 / 0.083	4.95 / 0.239	6.14 / 0.333	4.66 / 0.207
8	WavLM	1.14 / 0.032	0.26 / 0.006	1.20 / 0.034	2.47 / 0.121	4.34 / 0.216	1.91 / 0.060
Score fusion							
9	System 5 & 7	1.98 / 0.075	0.32 / 0.006	2.15 / 0.081	4.01 / 0.342	11.09 / 0.746	3.35 / 0.088
10	System 6 & 8	**1.10 / 0.0294**	**0.22 / 0.0051**	**1.16 / 0.0314**	**2.33 / 0.122**	**4.32 / 0.229**	**1.72 / 0.050**

4.1. Significance of self-supervised representations

Table 2 primarily presents the performance of four methods (TI-SV, TD-SV, Password rejection & TI-SV, and Password rejection & TD-SV), implemented with two architectures: ERes2Net and WavLM. The performance of systems 3 and 4 in Table 2 shows that the TD-SV system developed with the WavLM model outperforms the ERes2Net-based system. In the text-dependent evaluation, the WavLM model shows a relative improvement of 46.8 % and 43.2 % overall for Track 1 and Track 2, respectively, compared to the ERes2Net model. It highlights the significance of utilizing pretrained self-supervised models for text-dependent speaker verification tasks, especially with limited-scale training data.

4.2. Integrating password verification network

Since the baseline TD-SV system shows relatively low performance on target-wrong cases, we focus on improving this aspect. In Track 1 of the challenge, where the same ten passwords are used in training and evaluation, we can leverage a simple classification network for classification. Therefore, we developed a straightforward classification network, detailed in Section 3.1.1. Once trained, we compute the posterior probability of passwords to extract password information from the input speech signal. However, using posterior probabilities becomes less intuitive when evaluating unseen passwords. To extend the concept of password rejection to scenarios where unseen passwords are present in the evaluation, we employ an embedding comparison strategy to verify password information. Specifically, password embeddings extracted from the trained network are compared to ascertain if two signals share the same password.

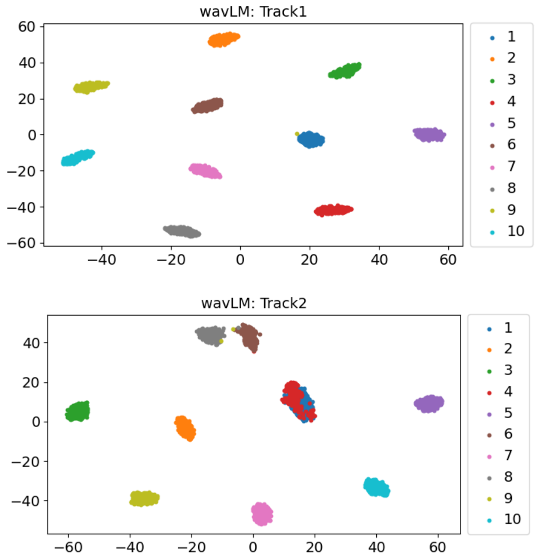

Fig. 2: t-SNE plots of password embeddings extracted from the sentence classifier. Track 2 - Passwords in the training data: [2,3,5,7,9,10], Passwords in the testing data: [1,4,6,8].

Our proposed method integrates this password verification module before the baseline system. Notably, the Track 1 password classifier achieves approximately 99 % accuracy on both training and development data. To visualize its performance, we analyzed 200 utterances of each password spoken by multiple speakers, producing t-SNE plots. The t-SNE plots of password embeddings extracted from Track 1's password classifier are depicted in Figure 2 (top). Given the clear separability of password embeddings, integrating this password rejection module ahead of the baseline system enhances overall performance. Results of this enhancement are presented in Table 2 under "Password rejection & TI-SV system" and "Password rejection & TD-SV system". As shown in Table 2, System 5 to 8, integration of the password rejection system notably improves performance in TC-vs-TW trials.

We also apply the password rejection module to Track 2 of the challenge. Despite Track 2 evaluating unseen passwords, the embeddings-based approach remains applicable. While integrating the password rejection module slightly improves Track 2 performance, its effectiveness is less pronounced than in Track 1. This could be attributed to the fact that the Track 2 password classifier is trained with six passwords and evaluated on four additional passwords not included in the training set. Upon inspecting the t-SNE plots (Figure 2) of password embeddings extracted from the Track 2 password classifier, we observe significant overlap among embeddings of unseen passwords. Among the 4 unseen passwords, the embeddings of passwords 1 and 4 are particularly indistinguishable due to their similarity (4 out of 6 words are the same) [1]. In contrast, passwords 6 and 8 are quite different, and their embeddings are clearly separable [2].

4.3. Final submission to the leaderboard

We performed score fusion of Systems 6 & 8 and submitted the results to Track 1 and 2 leaderboards. While score fusion consistently improved performance over standalone systems in Track 1, we did not observe the same trend in Track 2. Therefore, we submitted the scores of our best standalone system, System 8, to the Track 2 leaderboard. Our system ranked 1st in Track 1, but in Track 2, it ranked 3rd.

4.4. On the usage of free-text

Free-text spoken by the speaker is provided as training data for both tracks, and it is recommended to explore the potential use of free-text in developing the TD-SV system. As illustrated in Figure 1, our TD speaker embedding extractor is designed to classify both speaker and speaker-sentence labels. We use free-text speech data to classify speakers and password data to classify speaker-sentence classes. Table 3 presents the performance of the TD-SV system developed using the WavLM architecture, both with and without free-text speech data in training. We observe that incorporating free-text speech in training improves the performance of Track 1 but degrades the performance of Track 2. This may be because using free-text speech in Track 2 clearly degrades the performance of Tc-vs-TW trials, as the model attempts to classify only seen passwords, whereas the evaluation data contains unseen passwords. It enhances the performance of Track 1 because the evaluation only includes seen passwords.

Table 3: Investigating the usage of free-text.

Track 1: WavLM TD-SV			
Free-text	Overall	TC-vs-TW	TC-vs-IC
yes	**1.58 / 0.044**	**2.07 / 0.054**	**1.18 / 0.034**
no	1.97 / 0.0730	3.03 / 0.108	1.20 / 0.034
Track 2: WavLM TD-SV			
Free-text	Overall	TC-vs-TW	TC-vs-IC
yes	9.49 / 0.578	22.3 / 0.813	1.61 / 0.056
no	**3.22 / 0.161**	**6.25 / 0.263**	**1.56 / 0.056**

4.5. Analysis of pooling methods

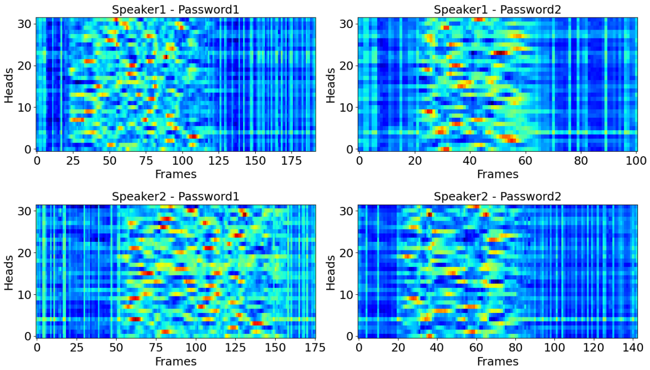

Fig. 3: Illustration of cluster assignments in MHFA. Password1: My voice is my password, Password2: Okay Google.

Table 4 presents the performance of Track 1 WavLM based TD-SV systems with different pooling methods, Multi-head attentive pooling (MHA) and Multi-head factorized attentive pooling (MHFA). As described in Section 3.1.2, MHFA pooling obtains cluster assignments using Key matrix $\mathbf{H}^k$. In contrast, MHA pooling uses the same frame-level representations ($\mathbf{H}^v$) for cluster assignments. As discussed in the literature [28], MHFA pooling outperforms MHA pooling. Figure 3 shows the plots of cluster assignment $\mathbf{A}$ in MHFA pooling. Here, the y-axis "Heads" represents the clusters. From the figures, we can see that cluster assignments are consistent for passwords across the speakers. This indicates that MHFA assigns frames to acoustic units and pools the representations within each acoustic unit.

Table 4: Performance of different pooling methods.

Pooling type	Overall	TC-vs-TW	TC-vs-IC
MHA	1.68	2.31	1.18
MHFA	**1.58**	**2.07**	**1.18**

[1] Password 1: sedaye man neshandahandeye hoviyyate man ast. Password 4: sedaye man ramze obure man ast.

[2] Password 6: My voice is my password. Password 8: Artificial intelligence is for real.

5. CONCLUSION

This paper provides a detailed description of team SIPLAB-IITH submission to the TDSV 2024 challenge for Track 1 and Track 2. It emphasizes the use of pretrained self-supervised representations, specifically the WavLM model, for text-dependent speaker verification tasks. The study also examines the integration of a password rejection module. While the password rejection module proves significant for Track 1 evaluation, its impact is minimal for Track 2, which involves unseen passwords in the evaluation. Finally, our submission to the challenge achieved 1st and 3rd place in Track 1 and Track 2, respectively. One potential area for future work could be developing a password rejection module that performs effectively even with unseen passwords.

6. ACKNOWLEDGMENT

This work was supported by DST National Mission Interdisciplinary Cyber-Physical Systems (NM-ICPS), Technology Innovation Hub on Autonomous Navigation and Data Acquisition Systems: TiHAN Foundations at Indian Institute of Technology (IIT) Hyderabad

7. REFERENCES

[1] Hossein Zeinali, Kong Aik Lee, Jahangir Alam, and Lukas Burget, “Short-duration speaker verification (sdsv) challenge 2021: the challenge evaluation plan,” *arXiv preprint arXiv:1912.06311*, 2019.

[2] Hossein Zeinali, Kong Aik Lee, Jahangir Alam, and Lukaš Burget, “Text-dependent speaker verification (tdsv) challenge 2024: Challenge evaluation plan.,” Tech. Rep., arXiv preprint arXiv:1xxx.0xxxx, 2024.

[3] Chee-Ming Ting, Sh-Hussain Salleh, Tian-Swee Tan, and AK Ariff, “Text independent speaker identification using gaussian mixture model,” in *2007 International Conference on Intelligent and Advanced Systems*. IEEE, 2007, pp. 194–198.

[4] Najim Dehak, Patrick J Kenny, Réda Dehak, Pierre Dumouchel, and Pierre Ouellet, “Front-end factor analysis for speaker verification,” *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 19, no. 4, pp. 788–798, 2010.

[5] David Snyder, Daniel Garcia-Romero, Daniel Povey, and Sanjeev Khudanpur, “Deep neural network embeddings for text-independent speaker verification.,” in *Interspeech*, 2017, vol. 2017, pp. 999–1003.

[6] David Snyder, Daniel Garcia-Romero, Gregory Sell, Daniel Povey, and Sanjeev Khudanpur, “X-vectors: Robust dnn embeddings for speaker recognition,” in *2018 IEEE international conference on acoustics, speech and signal processing (ICASSP)*. IEEE, 2018, pp. 5329–5333.

[7] Joon Son Chung, Jaesung Huh, Seongkyu Mun, Minjae Lee, Hee Soo Heo, Soyeon Choe, Chiheon Ham, Sunghwan Jung, Bong-Jin Lee, and Icksang Han, “In defence of metric learning for speaker recognition,” *arXiv preprint arXiv:2003.11982*, 2020.

[8] Brecht Desplanques, Jenthe Thienpondt, and Kris Demuynck, “Ecapa-tdnn: Emphasized channel attention, propagation and aggregation in tdnn based speaker verification,” *arXiv preprint arXiv:2005.07143*, 2020.

[9] Yafeng Chen, Siqi Zheng, Hui Wang, Luyao Cheng, Qian Chen, and Jiajun Qi, “An enhanced res2net with local and global feature fusion for speaker verification,” *arXiv preprint arXiv:2305.12838*, 2023.

[10] Arsha Nagrani, Joon Son Chung, and Andrew Zisserman, “Voxceleb: a large-scale speaker identification dataset,” *arXiv preprint arXiv:1706.08612*, 2017.

[11] Joon Son Chung, Arsha Nagrani, and Andrew Zisserman, “Voxceleb2: Deep speaker recognition,” *arXiv preprint arXiv:1806.05622*, 2018.

[12] Yue Fan, JW Kang, LT Li, KC Li, HL Chen, ST Cheng, PY Zhang, ZY Zhou, YQ Cai, and Dong Wang, “Cn-celeb: a challenging chinese speaker recognition dataset,” in *ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2020, pp. 7604–7608.

[13] Kong Aik Lee, Anthony Larcher, Guangsen Wang, Patrick Kenny, Niko Brümmer, David Van Leeuwen, Hagai Aronowitz, Marcel Kockmann, Carlos Vaquero, Bin Ma, et al., “The reddots data collection for speaker recognition,” in *Interspeech 2015*, 2015.

[14] Anthony Larcher, Kong Aik Lee, Bin Ma, and Haizhou Li, “The rsr2015: Database for text-dependent speaker verification using multiple pass-phrases,” in *Annual Conference of the International Speech Communication Association (Interspeech)*, 2012.

[15] Hossein Zeinali, Hossein Sameti, Lukas Burget, Jan Cernockỳ, Nooshin Maghsoodi, and Pavel Matejka, “i-vector/hmm based text-dependent speaker verification system for reddots challenge.,” in *InterSpeech*, 2016, pp. 440–444.

[16] Achintya Kumar Sarkar and Zheng-Hua Tan, “Text dependent speaker verification using un-supervised hmm-ubm and temporal gmm-ubm.,” in *interspeech*, 2016, pp. 425–429.

[17] Ehsan Variani, Xin Lei, Erik McDermott, Ignacio Lopez Moreno, and Javier Gonzalez-Dominguez, "Deep neural networks for small footprint text-dependent speaker verification," in *2014 IEEE international conference on acoustics, speech and signal processing (ICASSP)*. IEEE, 2014, pp. 4052–4056.

[18] Xiaoyi Qin, Danwei Cai, and Ming Li, "Far-field end-to-end text-dependent speaker verification based on mixed training data with transfer learning and enrollment data augmentation.," in *Interspeech*, 2019, pp. 4045–4049.

[19] Yichi Zhang, Meng Yu, Na Li, Chengzhu Yu, Jia Cui, and Dong Yu, "Seq2seq attentional siamese neural networks for text-dependent speaker verification," in *ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2019, pp. 6131–6135.

[20] Zhuxin Chen and Yue Lin, "Improving x-vector and plda for text-dependent speaker verification.," in *INTERSPEECH*, 2020, pp. 726–730.

[21] Bing Han, Zhengyang Chen, Zhikai Zhou, and Yanmin Qian, "The sjtu system for short-duration speaker verification challenge 2021," *arXiv preprint arXiv:2208.01933*, 2022.

[22] Steffen Schneider, Alexei Baevski, Ronan Collobert, and Michael Auli, "wav2vec: Unsupervised pre-training for speech recognition," *arXiv preprint arXiv:1904.05862*, 2019.

[23] Alexei Baevski, Yuhao Zhou, Abdelrahman Mohamed, and Michael Auli, "wav2vec 2.0: A framework for self-supervised learning of speech representations," *Advances in neural information processing systems*, vol. 33, pp. 12449–12460, 2020.

[24] Chengyi Wang, Yu Wu, Yao Qian, Kenichi Kumatani, Shujie Liu, Furu Wei, Michael Zeng, and Xuedong Huang, "Unispeech: Unified speech representation learning with labeled and unlabeled data," in *International Conference on Machine Learning*. PMLR, 2021, pp. 10937–10947.

[25] Sanyuan Chen, Chengyi Wang, Zhengyang Chen, Yu Wu, Shujie Liu, Zhuo Chen, Jinyu Li, Naoyuki Kanda, Takuya Yoshioka, Xiong Xiao, et al., "Wavlm: Large-scale self-supervised pre-training for full stack speech processing," *IEEE Journal of Selected Topics in Signal Processing*, vol. 16, no. 6, pp. 1505–1518, 2022.

[26] Shu-wen Yang, Po-Han Chi, Yung-Sung Chuang, Cheng-I Jeff Lai, Kushal Lakhotia, Yist Y Lin, Andy T Liu, Jiatong Shi, Xuankai Chang, Guan-Ting Lin, et al., "Superb: Speech processing universal performance benchmark," *arXiv preprint arXiv:2105.01051*, 2021.

[27] Shang-Hua Gao, Ming-Ming Cheng, Kai Zhao, Xin-Yu Zhang, Ming-Hsuan Yang, and Philip Torr, "Res2net: A new multi-scale backbone architecture," *IEEE transactions on pattern analysis and machine intelligence*, vol. 43, no. 2, pp. 652–662, 2019.

[28] Junyi Peng, Oldřich Plchot, Themos Stafylakis, Ladislav Mošner, Lukáš Burget, and Jan Černockỳ, "An attention-based backend allowing efficient fine-tuning of transformer models for speaker verification," in *2022 IEEE Spoken Language Technology Workshop (SLT)*. IEEE, 2023, pp. 555–562.

[29] Yingke Zhu, Tom Ko, David Snyder, Brian Mak, and Daniel Povey, "Self-attentive speaker embeddings for text-independent speaker verification.," in *Interspeech*, 2018, vol. 2018, pp. 3573–3577.

[30] Yingke Zhu and Brian Mak, "Bayesian self-attentive speaker embeddings for text-independent speaker verification," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 31, pp. 1000–1012, 2023.

[31] Hossein Zeinali, Hossein Sameti, and Themos Stafylakis, "Deepmine speech processing database: Text-dependent and independent speaker verification and speech recognition in persian and english.," in *Odyssey*, 2018, pp. 386–392.

[32] Hossein Zeinali, Lukáš Burget, and Jan Honza Černockỳ, "A multi purpose and large scale speech corpus in persian and english for speaker and speech recognition: the deepmine database," in *2019 IEEE Automatic Speech Recognition and Understanding Workshop (ASRU)*. IEEE, 2019, pp. 397–402.

[33] Jiankang Deng, Jia Guo, Niannan Xue, and Stefanos Zafeiriou, "Arcface: Additive angular margin loss for deep face recognition," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2019, pp. 4690–4699.